



Functional Safety Engineering

Safety Instrumented System Design and Development

Slide 5 - 1



Functional Safety Engineering

Achieving the target SIL

- Selection of Components and Sub Systems
- Design to achieve the target PFD average
- Design for safe behaviour on detection of a fault
- Ensure functional independence from BPCS
- Comply with fault tolerance requirements
- Design to reduce common cause failures
- Provide secure interfaces between components

ProSalus Limited

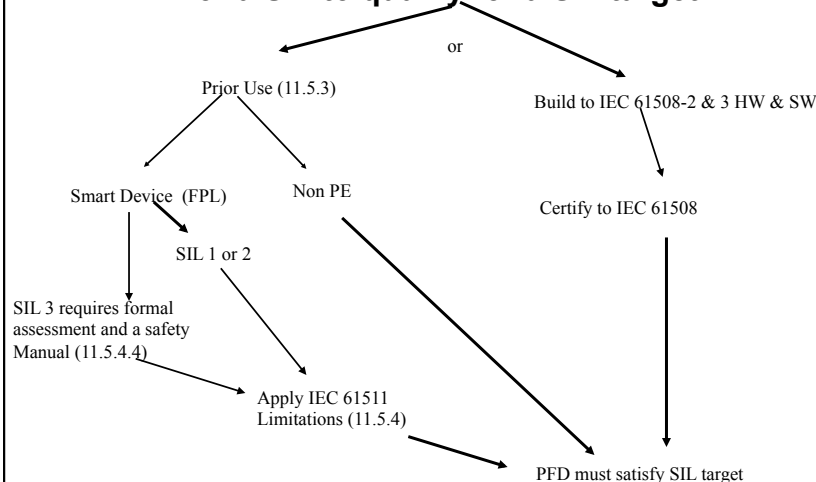
Slide 5 - 2

Selection of Components and Subsystems

Two paths to Functional Safety Compliance:

1. All components and subsystems in the SIF loop are designed and tested in accordance with IEC 61508-2/3
- OR
2. Evidence based on IEC 61511 “Prior Use” to demonstrate suitability of the SIF for a maximum target SIL2

For a SIF to qualify for a SIL target



Requirements for Device to be IEC 61511 “Proven-in-use”

- Evidence that the instrument is suitable for SIF
- Consider manufacturer's QA systems
- PES devices need formal validation –
IEC 61508-3 Annex A Table A.7 as starting point
- Performance record in a similar profile
- Adequate documentation
- Volume of experience, > 1 yr exposure per case.

Collect the records of every fault, failure, inspection, proof test, partial test and maintenance event per instrument.

The approved safety instrument list

- Each instrument that is suitable for SIF
- Update and monitor the list regularly
- Add instruments only when the data is adequate
- Remove instruments from the list when they let you down
- Adequate details: Include the process application

Managed by maintenance team and data fed to procurement



Functional Safety Engineering

Selection of Components and Subsystems

• IEC 61508 General requirements _

- Component developed to relevant IEC 61508 Part 2 & 3 requirements
- Safety Manual provided for specific component IEC 61508-2 Annex D
 - Functional Specification, Hardware / software configuration
 - Constraints and limitations on use identified during analysis (FMEDA)
 - Failure Modes for device and device diagnostics (Specifically those device failure modes not detected by diagnostics)
 - Failure Rates and Hardware Fault Tolerance
 - Type classification A or B, Systematic capability
 - Proof test, operating and maintenance requirements.
 - Calibration and set up features identified.

Slide 5 - 7



Functional Safety Engineering

Selection of Components and Subsystems

• Field Devices

- 'An initiator or final element used as part of a SIS shall not be used for control purposes where failure of the control system would cause a demand on the protection system except when an analysis has been carried out to confirm that the risk is acceptable'
- De-energize to trip is the preferred action.
- Energize to trip shall apply a continuous end-of-line monitor such as pilot current to ensure continuity.
- Smart sensors shall have write protection enabled.
- Must be suitable for the installed environment
 - i.e. Corrosion, temperature, humidity etc.

ProSalus Limited

Slide 5 - 8



Functional Safety Engineering

Sharing of Sensors with BPCS

When possible do not share sensors because it:

- Violates the principles of independence
- Potential for a high level of common mode failure
- Cannot not be considered a separate layer of protection
- Creates maintenance and change control issues

Separation Rules: Field Sensors IEC 61511 Part 1 : 11.2.4

ProSalus Limited

Slide 5 - 9



Functional Safety Engineering

Selection of Components and Subsystems

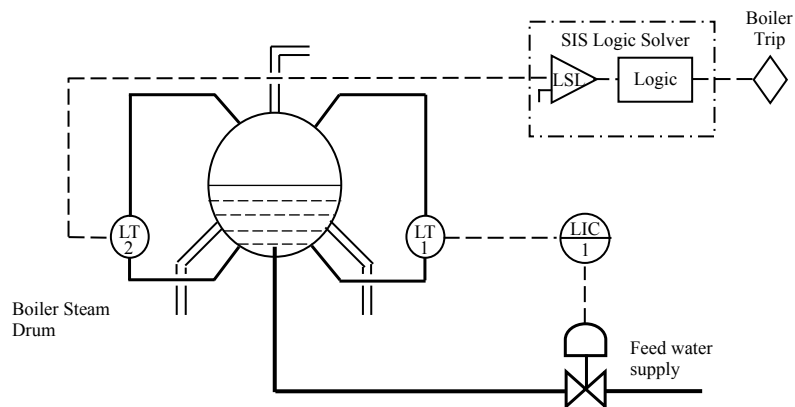
• Field sensors

- If a sensor is used for both BPCS and SIS then common mode failure considerations must be assessed
- Sensor diagnostics must be capable of placing the process in a safe state if a CMF occurs
- The Hardware Fault Tolerance requirements are met
- Separate sensors with identical or diverse redundancy will normally be required for SIL 3 & SIL 4 depending on the SFF.
- If SIS sensors are connected to a BPCS suitable isolator / splitters must be used and meet the target SIL requirements.

ProSalus Limited

Slide 5 - 10

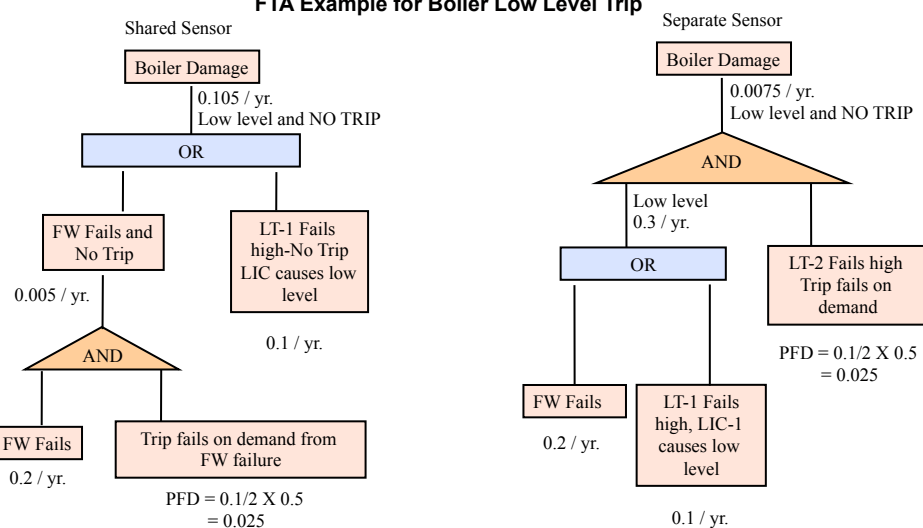
Separate Sensors for Control and Trip FTA Example



ProSalus Limited

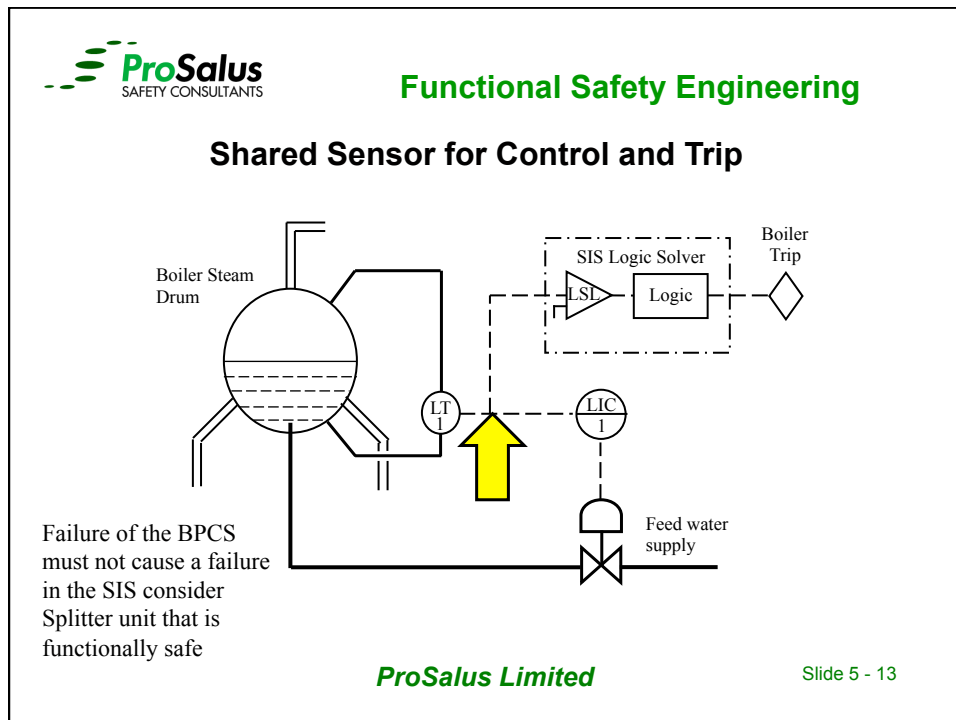
Slide 5 - 11

FTA Example for Boiler Low Level Trip



ProSalus Limited

Slide 5 - 12



ProSalus
SAFETY CONSULTANTS

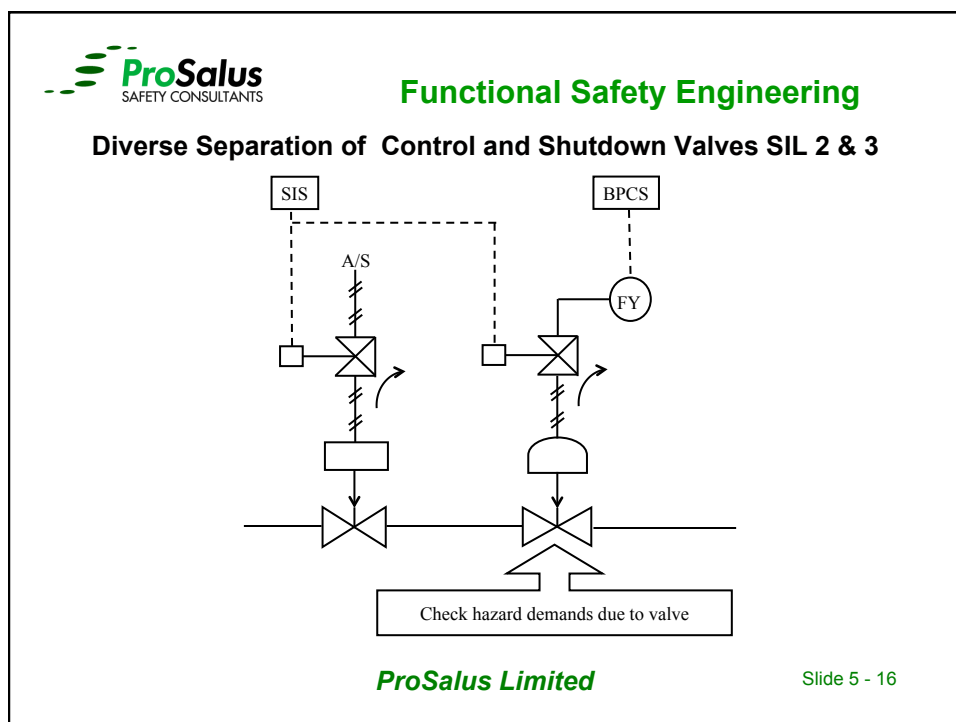
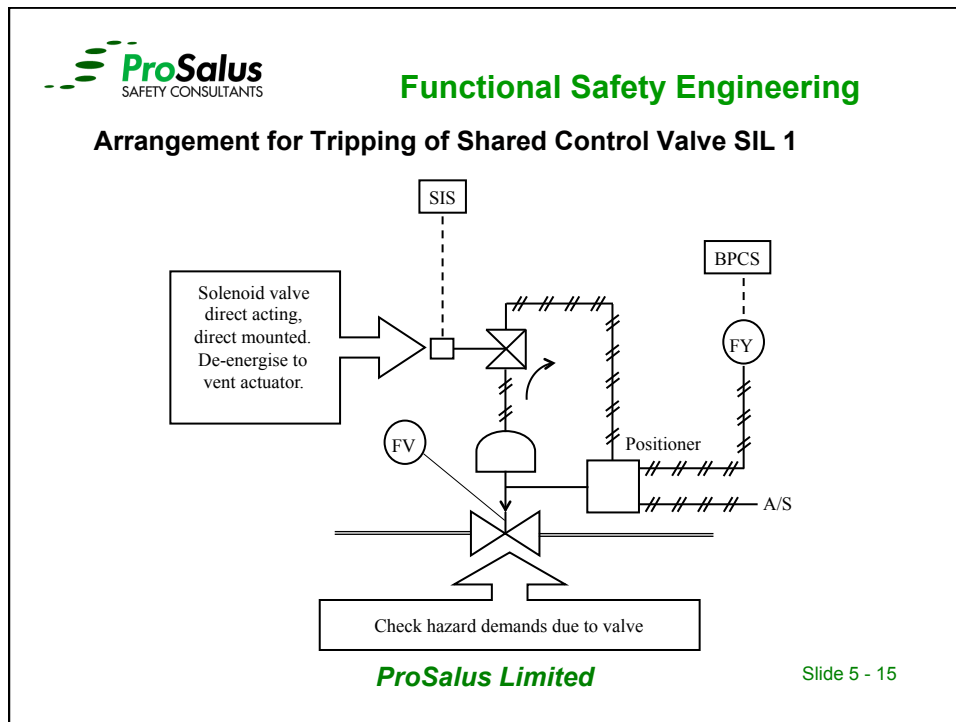
Functional Safety Engineering

Selection of Components and Subsystems

- **Control and shutdown valves**
 - A single valve may be used for both BPCS and SIS provided that:
 - A failure of the valve cannot cause a demand on the SIF
 - Diagnostic coverage on the valve and SIF will ensure safe reaction to a dangerous failure and common mode failure requirements are met.
 - Hardware Fault Tolerance requirements are met
 - SIL 3 and SIL 4 will normally require separate identical or diverse valves

ProSalus Limited

Slide 5 - 14





Functional Safety Engineering

Selection of Components and Subsystems

- **SIS Logic solver**
 - Functional separation between BPCS and SIS
 - Will have internal diagnostics to detect dangerous faults
 - Can be PES, Solid State or Relay
 - When there are a large number of outputs then it shall be necessary to determine if any foreseeable failures or combination of failures can lead to an hazardous event

ProSalus Limited

Slide 5 - 17



Functional Safety Engineering

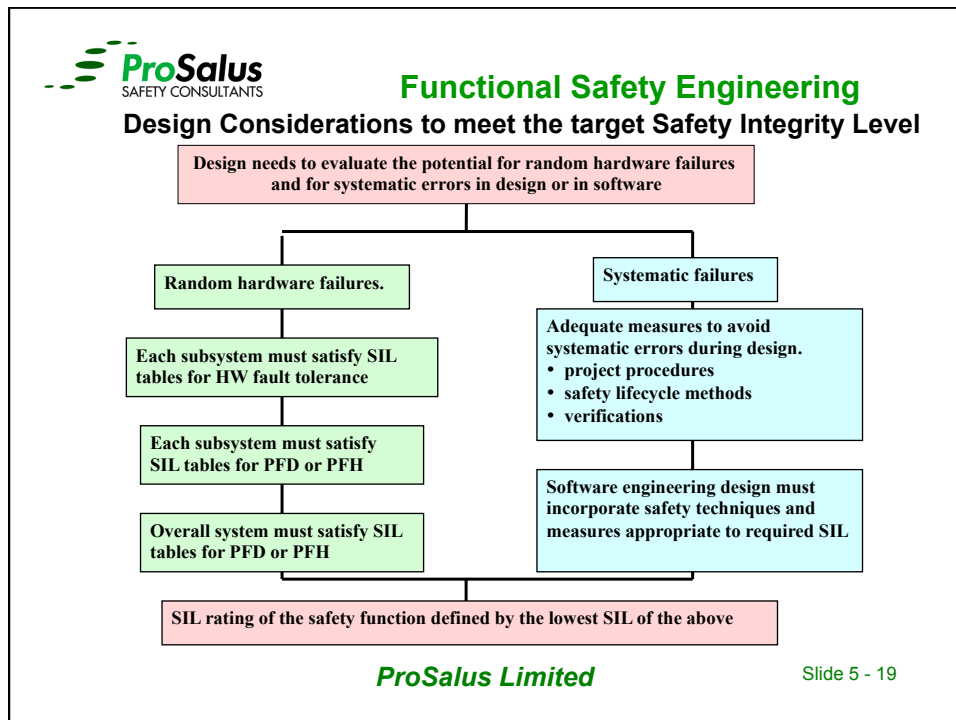
Design Considerations


Types of Failure

- The Integrity of a SIF is dependent on how often it fails dangerously.
- There are two main types of failure which need to be addressed:
 - Systematic failures;
 - Random hardware failures

ProSalus Limited

Slide 5 - 18



 **Functional Safety Engineering**
Systematic Failures

- A failure related in a deterministic way to a certain cause, which can only be eliminated by a modification of the design or the manufacturing process, operational procedures, documentation or other relevant factors:
 - Safety requirements specification;
 - Design;
 - Manufacture;
 - Installation;
 - Operation;
 - Maintenance
- Usually due to a human error, design fault-wrong component, incorrect specification error in software program, error in testing.

ProSalus Limited Slide 5 - 20



Functional Safety Engineering

Systematic failures

- A single systematic fault can cause failure in multiple channels of a redundant system.
- Systematic failures, by their very nature, cannot be accurately predicted because the events leading to them cannot be easily predicted.
- Functional safety standards protect against systematic faults providing rules, methods and guidelines to prevent design errors.
- A system implemented using such methods should be relatively free of systematic errors.

ProSalus Limited

Slide 5 - 21



Functional Safety Engineering

Random Failures

- A failure occurring at a random time, which results from one or more of the possible component degradation mechanisms.
- Random failures rates can be predicted with reasonable accuracy depending on the quality of the data
 - E.g. Generic, Industrial or Site failure rate data
 - Safe failures
 - Dangerous failures

ProSalus Limited

Slide 5 - 22



Functional Safety Engineering

Random Failures

- **Failure Rate data:**
 - Number of failures per unit / component as either:
 - A constant failure rate;
 - An average failure rate over a period / mission time
- **Dangerous failures are those that prevent success when there is a demand:**
 - Fails to operate when required i.e. valve fails to close
 - Worse are dormant failures – undetected dangerous failures
 - Potential consequences due to failure to prevent hazard occurring
- **Safe failures are spurious or nuisance failures:**
 - Spurious or nuisance shutdown no demand from process to trip
 - Downtime due to fault detection and restart
 - Loss of production / profits

ProSalus Limited

Slide 5 - 23



Functional Safety Engineering

IEC 61508 / 61511 Modes of Operation

- Three modes to consider:
 - Low
 - High
 - Continuous
- Most process plant SIFs are 'low demand mode'

ProSalus Limited

Slide 5 - 24



Functional Safety Engineering

Demand Modes

- **Low demand mode:**
 - An infrequent demand rate on a protective system;
 - No greater than once per year
 - Use Probability of Failure of Demand average (PFD_{avg})
- **High demand:**
 - The demand rate is greater than once per year
 - Use average frequency of dangerous failure (PFH)
- **Continuous demand**
 - Dangerous failure will lead to a potential hazard without any further failure
 - Use average frequency of dangerous failure (PFH).

ProSalus Limited

Slide 5 - 25



Functional Safety Engineering

Demand modes

Demand mode-61511	Continuous mode - 61511	
Low demand - 61508	High demand - 61508	Continuous - 61508
Use PFD _{avg}	Use probability of failure per hour	Use probability of failure per hour
Take credit for proof testing	No credit for proof testing	No credit for proof testing
Take credit for automatic diagnostics	Take credit for automatic diagnostics	No credit for automatic diagnostics

ProSalus Limited

Slide 5 - 26



Functional Safety Engineering

Average Probability of Failure on Demand

- A statistical probability or chance that a system will not perform its intended function when demanded.
- Valid for 'low demand mode' operation only

Average frequency of dangerous failure

- The average frequency of a dangerous failure of system to perform the specified safety function over a given period (PFH).
- Valid for 'high demand and continuous mode' operation only
- When the system is the ultimate layer PFH is calculated from unreliability $F(t) = 1 - R(t)$ approximates to $F(t)/T$ & $1/MTTF$
- When the system is not the ultimate layer PFH is calculated from unavailability $U(t)$ and approximates to $1/MTBF$

Slide 5 - 27



Functional Safety Engineering

Understanding Types of Failures

For a low Demand System SIFs can fail in two ways:

- Dangerous failure (hidden, covert or un-revealed)
 - Loss of protective function, but not aware until demand
 - Failure rate can be reduced by hardware fault tolerance (e.g. 1oo2 or 1oo3)
 - Diagnostics can also be used.
- Safe failure (revealed, evident – mostly economic)
 - Spurious or nuisance trip or alarm
 - No loss of protection
 - Spurious failures can be reduced by "revealed failure robustness" (e.g. 2oo2 or 2oo3)

ProSalus Limited

Slide 5 - 28

Which type of failures have impact on the SIF?

Many failures do not influence the safety function at all, and so they are not considered anymore. Example: Display, Keypad, HART communication).

In safety engineering we need to differentiate between safe and dangerous failures, and, if they are detectable or not (undetectable).

Safe failures impact on the SIF's availability, but not on the safety function.

Dangerous failures are split into detected and undetected.

Dangerous detected failures are detectable by diagnostic and will raise a diagnostic alarm or trip system into a safe state.

Slide 5 - 29

Understanding Types of Failure

	(Detectable)	(Undetectable)
(Safe)	λ_{SD}	λ_{SU}
(Dangerous)	λ_{DD}	λ_{DU}

Depending on the kind of evaluation safe and detectable failures can force a system to go into the safe state.

λ_{SD}	Signal cable to transmitter cut, Signal is 0 mA, central controller detects it: Safe!	λ_{SU}	Output transistor becomes defective, signal ≥ 20 mA, central controller triggers (maintaining) gas alarm: Safe!
λ_{DD}	Dangerous RAM-failure, being detected during automatic cyclic RAM-test, controller detects failure: Safe!	λ_{DU}	Loss of measuring function without indication: Unsafe – dangerous!

The Dangerous Undetectable failure (λ_{DU}) is in the main focus of the SIL-consideration.

Slide 5 - 30



Functional Safety Engineering

Design Considerations

- Improving reliability and integrity
 - Hardware fault tolerance;
 - Multiple devices:
 - 1oo2, 1oo3 etc.
- Avoidance of nuisance or spurious trips:
 - Voted multiple devices
 - 2oo2, 2oo3 etc.

ProSalus Limited

Slide 5 - 31



Functional Safety Engineering

Diagnostic Capability

- Ability of a sub system to automatically detect dangerous failures and take a action by:
 - Bringing the process to a safe state
 - Alerting the operator to take action – the diagnostic alarm should be included in the SIS in this case
- Thus when considering dangerous failures:
 - λ_{dd} = those dangerous failures that are detected by diagnostics:
 - λ_{du} = those dangerous failures that remain undetected by diagnostics and are only detected during Proof Testing

ProSalus Limited

Slide 5 - 32

Diagnostic Coverage

- The Diagnostic Coverage (DC) of a component or sub system is defined as the ratio of the average rate of dangerous detected failures of the component or sub system to the total average dangerous failure rate of the the component or sub system
- DC normally determined by FMEDA
- For pre certified or pre approved equipment the DC is included on the certificate of conformance

$$\text{Diagnostic Coverage} = \frac{\sum \lambda_{DD}}{\sum \lambda_{DD} + \sum \lambda_{DU}}$$

Design Considerations - Sensor Diagnostics

- Do not confuse with proof testing
- Integral to the device, designed in after OEM FMEDA has been completed to determine potential diagnostic mechanisms
- Must ensure diagnostic output is used and either trips the SIF or operator is trained to understand requirements of diagnostic alarms or NO credit for diagnostics should be taken in calculations
- Could compare trip transmitter value with related variables when practicable but not a secure method and puts more pressure on operator
- Diagnostic alarm test must be included in proof test to ensure operator awareness stays high

Valve Diagnostics

Failure Mode	% Contribution to dangerous failures	%Detection by partial closure test	% Of Dangerous Faults Detected
Actuator spring breakage or jamming	20	70	14
Solenoid fails to vent	5	50	2.5
Positioner fails to trip	5	100	5
Hoses kinked or blocked	10	100	10
Valve stem or rotary shaft stuck	40	70	28
Actuator linkage fault	5	70	3.5
Seating failures of valve causing high leakage. Due to erosion or corrosion	10	0	0
Foreign bodies or sludge preventing full closure	5	0	0
Total	100%		63%

Methods for Valve Diagnostics

- On-line functional testing
- Limit switch discrepancy / mismatch alarm
- Position feedback
- Partial closure testing – manual or automatic
- Smart Positioner – certified safety Positioner

Architectural Constraints

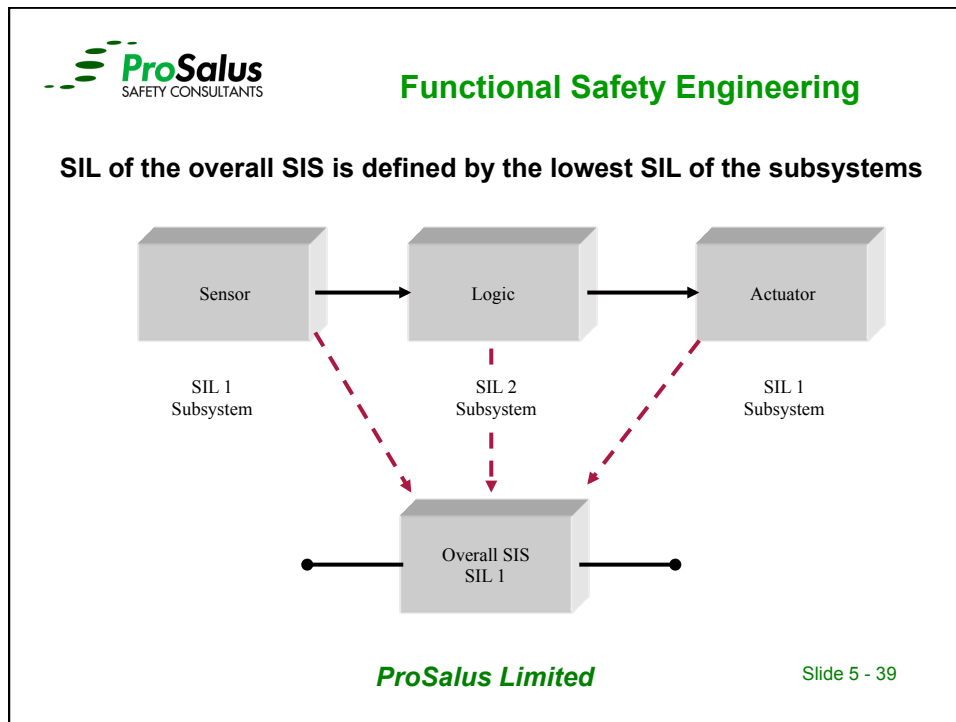
Subsystem Safety Integrity

Architectural Constraints and Hardware Fault Tolerance

Hardware Fault tolerance:

Hardware fault tolerance is the ability of a system to continue to be able to undertake the required safety function in the presence of one or more dangerous faults in hardware. Hence a fault tolerance level of 1 means that a single dangerous fault in the equipment will not prevent the system from performing its safety functions.

From the above it follows that a fault tolerance level of zero implies that the system cannot protect the process if a single dangerous fault occurs in the equipment.



ProSalus
SAFETY CONSULTANTS

Functional Safety Engineering

Safe Failure Fraction

- The Safe Failure Fraction (SFF) of a sub system is defined as the ratio of the average rate of safe plus dangerous detected failures of the sub system to the total average failure rate of the sub system
- SFF normally determined by FMEDA
- For pre certified or pre approved equipment the SFF is included on the certificate of conformance

$$\text{Safe Failure Fraction} = \frac{\sum \lambda_S + \sum \lambda_{DD}}{\sum \lambda_S + \sum \lambda_{DD} + \sum \lambda_{DU}}$$

ProSalus Limited Slide 5 - 40



Functional Safety Engineering

Architectural Constraints

- IEC 61508 places an upper limit on the SIL that can be claimed for any SIF on the basis of the HFT of the subsystems that it uses.
- Limit is a function of
 - Device Type A or B
 - The degree of confidence in the behaviour under fault conditions
 - Safe Failure Fraction
 - Hardware fault tolerance

ProSalus Limited

Slide 5 - 41



Functional Safety Engineering

IEC 61508 Classification of Equipment

IEC 61508 defines two types of equipment for use in SIS:

- Type A: Simple Devices: Non PES – where failure modes and fault behaviour are well defined and there is dependable failure data
- Type B: Complex Devices: Including PES - where failure modes and fault behaviour are not well defined and there is insufficient dependable failure data
- Fault tolerance rating of B is less than A for equivalent SFF

ProSalus Limited

Slide 5 - 42



Functional Safety Engineering

IEC 61508 Table 2

Minimum hardware fault tolerance of type A sub systems

SIL	Minimum HW Fault Tolerance			
	SFF<60%	SFF 60% to 90%	SFF>90%	SFF>99%
1	0	0	0	0
2	1	0	0	0
3	2	1	0	0
4		2	1	1

**For devices with well defined failure modes,
predictable behaviour and field experience.
Normally excludes PES**

ProSalus Limited

Slide 5 - 43



Functional Safety Engineering

IEC 61508 Table 3

Minimum hardware fault tolerance of type B sub systems

SIL	Minimum HW Fault Tolerance			
	SFF<60%	SFF 60% to 90%	SFF>90%	SFF>99%
1	1	0	0	0
2	2	1	0	0
3		2	1	0
4			2	1

**For devices with some none defined failure modes
OR unpredictable behaviour
OR insufficient field experience**

ProSalus Limited

Slide 5 - 44



Functional Safety Engineering

IEC 61511-1 Table 6
Minimum hardware fault tolerance of sensors, final elements & non PES logic

SIL	Minimum HW Fault Tolerance
1	0
2	1
3	2
4	Special requirements: See IEC 61508

The following summarized conditions apply for SIL 1,2 and 3 :

Increase FT by 1 if instrument does not have fail safe characteristics

Decrease FT by 1 if instrument if the device complies with the following.

- The hardware is selected on the basis of prior use (IEC 61511 11.5.3)
- The device allows adjustment of process related parameters only, for example, measuring range, upscale or downscale failure detection.
- The adjustment of the process related parameters of the device is protected, for example jumper, password.
- The function has a SIL requirement of less than 4.

Alternatively tables 2 and 3 of IEC 61508 may be applied if the SFF can be calculated

ProSalus Limited

Slide 5 - 45



Functional Safety Engineering

Architecture rules for PES logic solvers
IEC 61511-1 Table 5
Minimum hardware fault tolerance of PE logic solvers

SIL	Minimum HW Fault Tolerance		
	SFF<60%	SFF 60% to 90%	SFF>90%
1	1	0	0
2	2	1	0
3	3	2	1
4	Special requirements: See IEC 61508		

Alternatively tables 2 and 3 of IEC 61508 may be applied with an assessment

ProSalus Limited

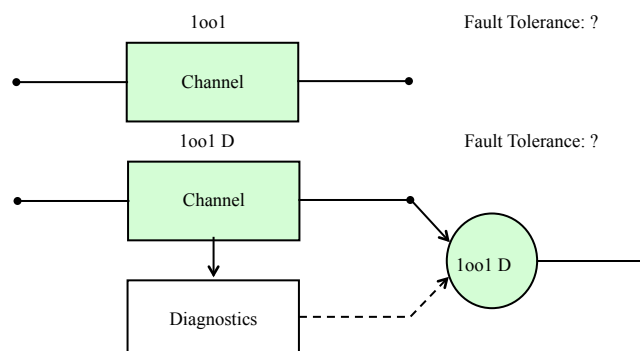
Slide 5 - 46

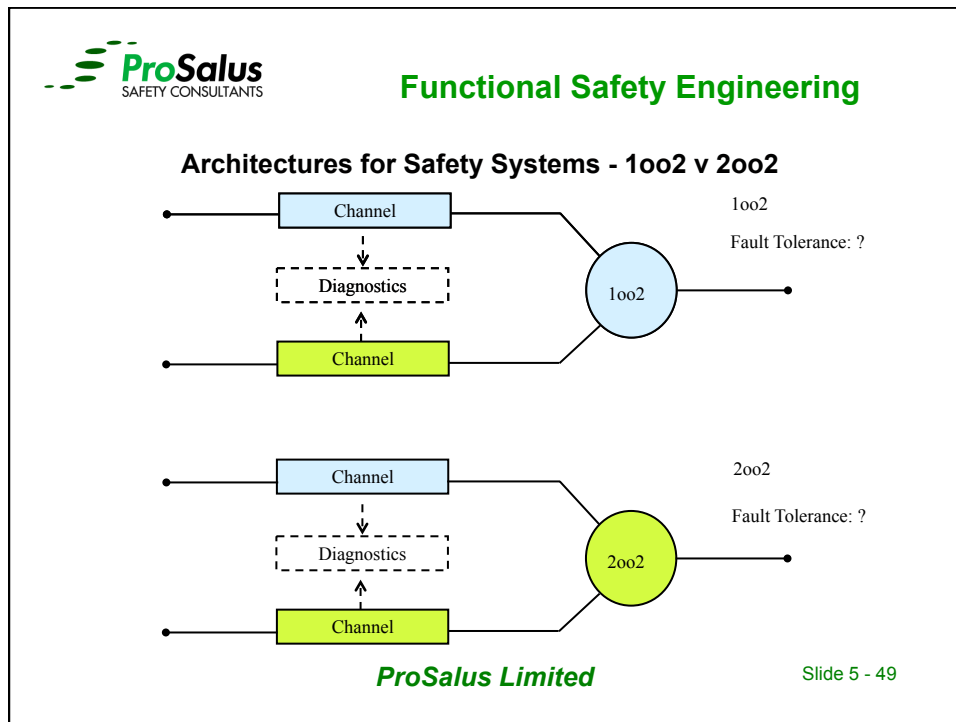
Example Minimum Architectures for Fault Tolerance of Type A and B Sub-Systems for 60% to 90% SFF

Safety Integrity.	Simple Devices (Non PES) Type A		Complex Devices Type B	
	Min. Fault tolerance.	Minimum Architecture	Min. Fault tolerance.	Minimum Architecture
SIL 1	0	1oo1	0	1oo1
SIL 2	0	1oo1	1	1oo2 or 2oo3
SIL 3	1	1oo2 or 2oo3	2	1oo3
SIL 4	2	1oo3	Special requirements apply, see IEC 61508	

Architectures for Safety Systems - 1oo1 Single channel

1oo1D has a higher safe failure fraction than 1oo1 but is still not able to protect the plant if a fault remains hidden





ProSalus
SAFETY CONSULTANTS

Functional Safety Engineering

Performance attributes of sub-system architectures

Sub system structure	Fault tolerance	Selection Guide
1oo1	0	Use if both PFD and nuisance trip targets are met.
1oo2	1	2 Sensors installed, 1 required to trip. PFD value improved, nuisance trip rate doubled. Often suitable for SIL 2
2oo3	1	3 Sensors installed, 2 required to trip. PFD improved over 1oo1, nuisance trip rate dramatically reduced.
1oo1D	0	Internal and external diagnostics used to improve safe failure fraction. Alternative to 1oo2 for SIL2
1oo2D	1	As for 1oo1D but able to tolerate 1 fault and revert to 1oo1D during repair. Meets SIL 3 if safe failure fraction exceeds 90%. Does not satisfy diversity for SIL3 if sensors are identical. Reduces spurious trip rate, good alternative to 2oo3
1oo3	2	3 Sensors installed, 1 required to trip. PFD improved over 1oo2 but not by much unless diverse instruments are used. Nuisance trip rate may be a problem. Likely to be used for SIL 2 or 3.
2oo4	2	Configured as two voting pairs of 1oo2D. Very high performance when used in logic solvers. Achieves SIL 3 performance with 1 pair off line for repair.

ProSalus Limited

Slide 5 - 50



Functional Safety Engineering

Safe Failure Fraction - Issues

- Optimistic claims for dangerous failures that can be detected by diagnostics
- FMEDA is considered best practice for assessing dangerous failures that can be detected by diagnostics
- If the detected failure claim is too optimistic then the safety integrity will be compromised due to the reduction in Hardware fault tolerance

ProSalus Limited

Slide 5 - 51



Functional Safety Engineering

Common Cause & Common Mode Failures

- A CCF occurs when a single fault results in the corresponding failure of multiple components.
- Common mode failures are a subset of common cause failures'
- *"A common-mode failure (CMF) is the result of an event (s) which because of dependencies, causes a coincidence of failure states of components in two or more separate channels of a redundancy system, leading to the defined system failing to perform its intended function".*

ProSalus Limited

Slide 5 - 52

Common Cause Failures in Sensors

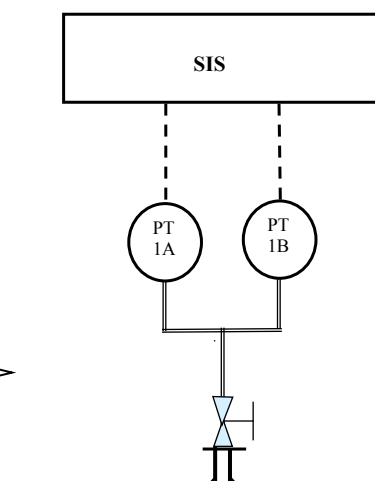
- Wrong specification
- Hardware design errors
- Software design errors
- Environmental stress
- Shared process connections
- Wrong maintenance procedures
- Incorrect calibration

ProSalus Limited

Slide 5 - 53

Design Issues: Redundancy in Sensors

Be careful to analyze for
common cause faults
eg. Try to avoid this



ProSalus Limited

Slide 5 - 54



Functional Safety Engineering

Common Cause Failures (IEC 61508)

- IEC 61508 Part 6 Annex D –Method for quantifying CCF
- 2010 version updated and based on PDS methodology
- Based on the following factors from IEC 61508-6 Table D.1 to D5:
 - Separation/segregation;
 - Diversity/redundancy;
 - Complexity/design/application/experience;
 - Assessment/analysis & feedback of data;
 - Procedures/human interface;
 - Competence/training/safety culture;
 - Environmental Control;
 - Environmental Testing.


ProSalus Limited

Slide 5 - 55


Table D.1 – Scoring programmable electronics or sensors/final elements



Item	Logic subsystem		Sensors and final elements	
	X	Y	X _s	Y _s
Separation/segregation				
Are all signal cables for the channels routed separately at all positions?	1,5	1,5	1,0	2,0
Are the logic subsystem channels on separate printed-circuit boards?	3,0	1,0		
Are the logic subsystem channels in separate cabinets?	2,5	0,5		
If the sensors/final elements have dedicated control electronics, is the electronics for each channel on separate printed-circuit boards?			2,5	1,5
If the sensors/final elements have dedicated control electronics, is the electronics for each channel indoors and separate cabinets?			2,5	0,5
Diversity/redundancy				
Do the channels employ different electrical technologies for example, one electronic or programmable electronic and the other relay?	7,0			
Do the channels employ different electrical technologies for example, one electronic, the other programmable electronic?	5,0			
Do the devices employ different physical principles for the sensing elements for example, pressure and temperature, vane anemometer and Doppler transducer, etc?			7,5	
Do the devices employ different electrical principles/designs for example, digital and analogue, different manufacturer (not re-badged) or different technology?			5,5	
Do the channels employ enhanced redundancy with MooN architecture where $N > M + 2$?	2,0	0,5	2,0	0,5
Do the channels employ enhanced redundancy with MooN architecture where $N = M + 2$?	1,0	0,5	1,0	0,5
Is low diversity used, for example hardware diagnostic tests using the same technology?	2,0	1,0		
Is medium diversity used, for example hardware diagnostic tests using different technology?	3,0	1,5		
Were the channels designed by different designers with no communication between them during the design activities?	1,0	1,0		
Are separate test methods and people used for each channel during commissioning?	1,0	0,5	1,0	1,0
Is maintenance on each channel carried out by different people at different times?	2,5		2,5	
Complexity/design/application/maturity/experience				
Does cross-connection between channels preclude the exchange of any information other than that used for diagnostic testing or voting purposes?	0,5	0,5	0,5	0,5
Is the design based on techniques used in equipment that has been used successfully in the field for > 5 years?	0,5	1,0	1,0	1,0
Is there more than 5 years experience with the same hardware used in similar environments?	1,0	1,5	1,5	1,5
Is the system simple, for example no more than 10 inputs or outputs per channel?		1,0		
Are inputs and outputs protected from potential levels of over-voltage and over-current?	1,5	0,5	1,5	0,5
Are all devices/components conservatively rated (for example, by a factor of 2 or more)?	2,0		2,0	
Assessment/analysis and feedback of data				
Have the results of the failure modes and effect analysis or fault-tree analysis been examined to establish sources of common cause failure and have predetermined sources of common cause failure been eliminated by design?		3,0		3,0
Were common cause failures considered in design reviews with the results fed back into the design? (Documentary evidence of the design review activity is required.)		3,0		3,0
Are all field failures fully analysed with feedback into the design (Documentary evidence of the procedure is required.)	0,5	3,5	0,5	3,5



Item	Logic subsystem		Sensors and final elements	
	X	Y	X _s	Y _s
Procedures/human interface				
Is there a written system of work to ensure that all component failures (or degradations) are detected, the root causes established and other similar items inspected for similar potential causes of failure?		1,5	0,5	1,5
Are procedures in place to ensure that: maintenance (including adjustment or calibration) of any part of the independent channels is staggered, and, in addition to the manual checks carried out following maintenance, the diagnostic tests are allowed to run satisfactorily between the completion of maintenance on one channel and the start of maintenance on another?	1,5	0,5	2,0	1,0
Do the documented maintenance procedures specify that all parts of redundant systems (for example, cables, etc.) intended to be independent of each other, are not to be relocated?	0,5	0,5	0,5	0,5
Is all maintenance of printed-circuit boards, etc. carried out off-site at a qualified repair centre and have all the repaired items gone through a full pre-installation testing?	0,5	1,0	0,5	1,5
Does the system have low diagnostics coverage (60 % to 90 %) and report failures to the level of a field-replaceable module?	0,5			
Does the system have medium diagnostics coverage (90 % to 99 %) and report failures to the level of a field-replaceable module?	1,5	1,0		
Does the system have high diagnostics coverage (>99 %) and report failures to the level of a field-replaceable module?	2,5	1,5		
Does the system diagnostic tests report failures to the level of a field-replaceable module?			1,0	1,0
Competence/training/safety culture				
Have designers been trained (with training documentation) to understand the causes and consequences of common cause failures?	2,0	3,0	2,0	3,0
Have maintainers been trained (with training documentation) to understand the causes and consequences of common cause failures?	0,5	4,5	0,5	4,5
Environmental control				
Is personnel access limited (for example locked cabinets, inaccessible position)?	0,5	2,5	0,5	2,5
Is the system likely to operate always within the range of temperature, humidity, corrosion, dust, vibration, etc., over which it has been tested, without the use of external environmental control?	3,0	1,0	3,0	1,0
Are all signal and power cables separate at all positions?	2,0	1,0	2,0	1,0
Environmental testing				
Has the system been tested for immunity to all relevant environmental influences (for example EMC, temperature, vibration, shock, humidity) to an appropriate level as specified in recognized standards?	10,0	10,0	10,0	10,0
<p>NOTE 1 A number of the items relate to the operation of the system, which may be difficult to predict at design time. In these cases, the designers should be make reasonable assumptions and subsequently ensure that the eventual user of the system is made aware of, for example, the procedures to be put in place in order to achieve the designed level of safety integrity. This could be by including the necessary information in the accompanying documentation.</p> <p>NOTE 2 The values in the X and Y columns are based on engineering judgement and take into account the indirect as well as direct effects of the items in column 1. For example, the use of field-replaceable modules leads to</p> <ul style="list-style-type: none"> - repairs being carried out by the manufacturer under controlled conditions instead of (possibly incorrect) repairs being made under less appropriate conditions in the field. This leads to a contribution in the Y column because the potential for systematic (and, hence, common cause) failures is reduced; - a reduction in the need for on-site manual interaction and the ability quickly to replace faulty modules, possibly on-line, so increasing the efficacy of the diagnostics for identifying failures before they become common-cause failures. This leads to a strong entry in the X column. 				



Functional Safety Engineering

Table D.2 – Value of Z: programmable electronics

Diagnostic coverage	Diagnostic test interval		
	Less than 1 min	Between 1 min and 5 min	Greater than 5 min
≥ 99 %	2,0	1,0	0
≥ 90 %	1,5	0,5	0
≥ 60 %	1,0	0	0

Table D.3 – Value of Z: sensors or final elements

Diagnostic coverage	Diagnostic test interval			
	Less than 2 h	Between 2 h and two days	Between 2 days and one week	Greater than one week
≥ 99 %	2,0	1,5	1,0	0
≥ 90 %	1,5	1,0	0,5	0
≥ 60 %	1,0	0,5	0	0

Table D.4 – Calculation of β_{int} or β_{out}

Score (S or S ₀)	Corresponding value of β or β_0 for the:	
	Logic subsystem	Sensors or final elements
120 or above	0,5 %	1 %
70 to 120	1 %	2 %
45 to 70	2 %	5 %
Less than 45	5 %	10 %

NOTE 1 The maximum levels of β_0 shown in this table are lower than would normally be used, reflecting the use of the techniques specified elsewhere in this standard for the reduction in the probability of systematic failures as a whole, and of common cause failures as a result of this.

NOTE 2 Values of β_0 lower than 0,5 % for the logic subsystem and 1 % for the sensors would be difficult to justify.

Slide 5 - 58

Table D.5 – Example values for programmable electronics

Category		Diverse system with good diagnostic testing	Diverse system with poor diagnostic testing	Redundancy system with good diagnostic testing	Redundancy with poor diagnostic testing
Separation/segregation	X	3,50	3,50	3,50	3,50
	Y	1,50	1,50	1,50	1,50
Diversity/redundancy	X	14,50	14,50	2,00	2,00
	Y	3,00	3,00	1,00	1,00
Complexity/design/....	X	2,75	2,75	2,75	2,75
	Y	2,25	2,25	2,25	2,25
Assessment/analysis/....	X	0,25	0,25	0,25	0,25
	Y	4,75	4,75	4,75	4,75
Procedures/human interface	X	3,50	3,50	3,50	3,50
	Y	3,00	3,00	3,00	3,00
Competence/training/....	X	1,25	1,25	1,25	1,25
	Y	3,75	3,75	3,75	3,75
Environmental control	X	2,75	2,75	2,75	2,75
	Y	2,25	2,25	2,25	2,25
Environmental test	X	5,00	5,00	5,00	5,00
	Y	5,00	5,00	5,00	5,00
Diagnostic coverage	Z	2,00	0,00	2,00	0,00
Total		33,5	33,5	21	21
Total Y		25,5	25,5	23,5	23,5
Score S		59	59	44,5	44,5
β_{int}		2 %	2 %	5 %	5 %
Score S _D		126	59	86,5	44,5
β_{Dint}		0,5 %	2 %	1 %	5 %

Slide 5 - 59

Common Cause Failures (IEC 61508)

- Using the IEC 61508 Part 6 Annex D β -factor model
 - Common Cause failure rate is $\lambda_D \beta$
 - Where diagnostics are available overall CCF rate is

$$\lambda_{DU} \beta + \lambda_{DD} \beta_D$$

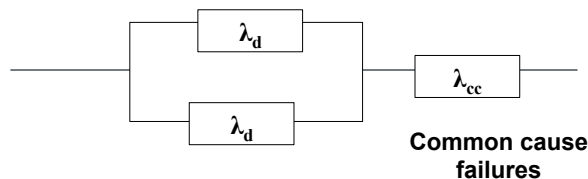
- Using Table D1, D2, D3 and D4
- β - $S = X + Y = \beta_{int}$ for a 1oo2 System
- β_D - $S_D = X (Z+1) + Y = \beta_{Dint}$ for a 1oo2 System

Common Cause Failures

- Apply Table D5 for systems with levels of redundancy greater than 1oo2 , table based on PDS Method
- IEC 61508-3, Annex D, Table D.4 for 1oo2
 - *0.01 –0.1 for field equipment;*
 - *0.005 –0.05 for programmable electronic systems*

Common Cause Failures – Systems Diagram 1oo2

- Consider a simple 1oo2 redundant subsystem



λ_d = total dangerous failure rate

λ_{cc} = total common cause failure rate

$\lambda_{cc} = \beta \lambda_d$

Where β = the common cause failure factor



Functional Safety Engineering

Common Cause Failure Calculation - Example

$$\lambda_{cc} = \lambda_{\text{common cause}}$$

$$\lambda_{cc} = \beta \lambda_d$$

Where:

$$\lambda_d = 0.05 \text{ failures / year}$$

$$\beta = 0.1$$

Therefore

$$\lambda_{cc} = 0.1 * 0.05 = 0.005 \text{ failures / year}$$

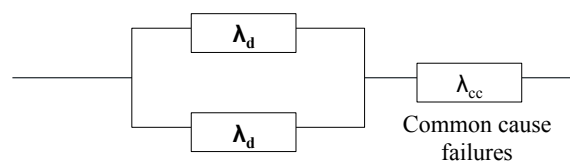
ProSalus Limited

Slide 5 - 63



Functional Safety Engineering

Common Cause Failures – Systems Diagram 1oo2



$$PFD_{avg} = \frac{(\lambda_d)^2 \times T^2}{3} + \frac{\lambda_{cc} \times T}{2}$$

Common Cause failure should be shown as an additional 1oo1 block in the RBD or as an input to an OR gate in an FTA and then summed with the 1oo2 block to calculate overall sub system PFD_{avg}

ProSalus Limited

Slide 5 - 64



Functional Safety Engineering

Design Considerations - Field Devices Summary

- Safety Related Instruments must well proven
- Smart instrumentation treated as PES – Type B
- Separation, Redundancy, Diversity design issues
- Increased Diagnostic Coverage for improved SFF to reduce HFT requirements
- For SIL 1 and SIL 2 - justification of suitability on “prior use”.
 - Requires evidence of previous usage in safety.
 - SIL 3 requires formal assessment (IEC 61511 11.5.4.4)
 - “Prior use” does not help if the instrument is new to your company unless the vendor can assist with Client data

ProSalus Limited

Slide 5 - 65



Functional Safety Engineering

Safety Component Selection

- Use Safety Certified / approved components to IEC 61508 wherever possible as this aids in the verification in terms of failure data, component type, safe failure fraction, available diagnostics
- Make sure Safety Manual is supplied with device / component.
- Ensure application and usage complies with vendor's safety manual.
- If you have records of the same instrument being used for an extensive period in safety applications you can document your own “Prior use” justification up to SIL 2 only.
- Insist on verifiable data from Vendor / system supplier for the device / component either based on FMEDA, returns data or accelerated testing.

ProSalus Limited

Slide 5 - 66

IEC 61511 Application Software

Slide 5 - 67

Software Safety Topics

- Software for Safety
- Software Verification & Systematic Errors
- Software Management & Quality Assurance
- Software Safety life cycle
- Software Safety Requirements
- Certification and compliance

ProSalus Limited

Slide 5 - 68



Functional Safety Engineering

Software for Safety

- SIS software must have a proven QA/C (Testing) and FSM record
- Software comes in two parts: Embedded and Application
 - Both parts require software QA/C & FS management procedures
 - Embedded software including development tools QA/C & FS management procedures and software construct should to be 3rd party certified to IEC 61508-3 with a report of limitations of use
 - Application tools should be certified for use with the OEM software package
 - Development of Application software to follow IEC 61511-1 figures 12 Software development lifecycle table 7 and comply with IEC 61131 software language requirements.

ProSalus Limited

Slide 5 - 69



Functional Safety Engineering

Software Verification & Systematic Errors

- IEC 61508-3 safety approved embedded / operating system and check versions are certified for use with hardware and application package
- IEC 61508-3 precertified software modules (Function Blocks)
- OEM approved application package matched to system hardware and software versions.
- IEC 61511 Clause 12 for QA and FSM procedures for application software when using IEC 61508 compliant systems
- IEC 61511 Software Validation by Testing against Requirement Specification and cause and effects
- Software verification complicated – 61508 requires formal analysis & traceability (61508-7 Annex D)

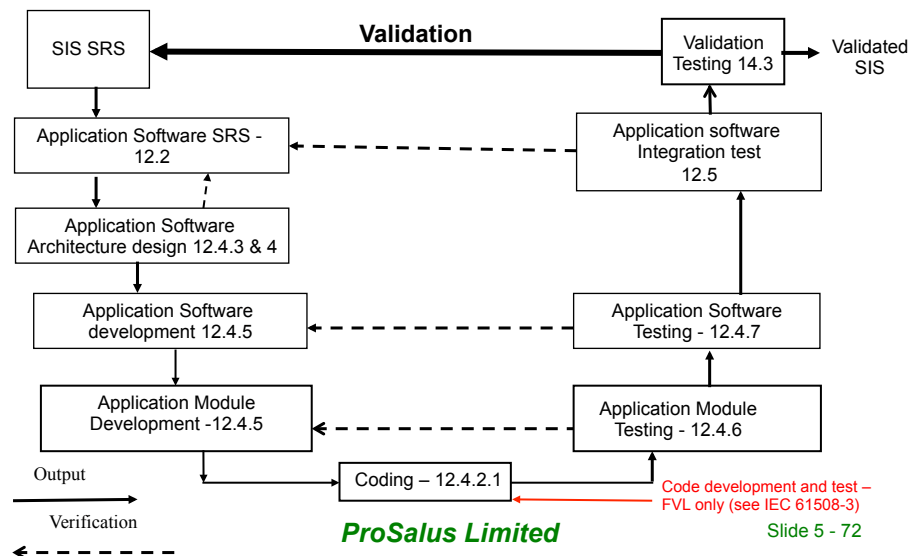
ProSalus Limited

Slide 5 - 70

Software Management and Quality Assurance

- Management of Software Quality & Testing replaces reliability analysis
- Software Quality Assurance practices are well established.
- IEC 61508-6 Annex E Safety Manual requirements for Software Elements
- Software Safety Life Cycle in IEC 61508-3 Annex G for detailed guidance on software lifecycles and IEC 61511 clause 12
- IEC 61508-6 Annex E for example guidance on the application of the IEC 61508-3 software safety integrity tables

Functional Safety Engineering Software Safety Lifecycle: V model





Functional Safety Engineering

Application Software Life Cycle Requirements

- Application Software Safety Requirements Specification
- Features and facilities required of the application language
- Features to facilitate safe modification of the application
- Architecture of the application software
- Requirements for support tools, user manual and application languages
- Software development methods
- Software module testing
- Software integration testing
- Integration testing with the SIS subsystem

Continues through to Validation, Operation, Proof testing and Inspection.

ProSalus Limited

Slide 5 - 73



Functional Safety Engineering

Specification of Application Software

Software safety requirements, in terms of both the safety functions and the safety integrity, should be stated in the safety requirements specification.

The specification should include all the modes of operation, the capacity and response time performance requirements, maintenance and operator requirements, self-monitoring of the software and hardware as appropriate, enabling the safety function to be testable whilst the plant is operational, and details of all internal/external interfaces.

The specification should be written in a clear and precise manner and should be free from ambiguity and clear to those for whom it is intended.

For SIL 2 systems, the specification should use semi-formal methods to describe the critical parts of the requirement (e.g. Safety related control logic). The semi-formal methods chosen should be appropriate to the application and typically include logic/function block diagrams, cause and effect charts, sequence diagrams, state transition diagrams, time Petri nets, truth tables and data flow diagrams.

ProSalus Limited

Slide 5 - 74



Functional Safety Engineering

Design and Development

The design methods should aid modularity and embrace features which reduce complexity and provide clear expression of functionality, information flow, data structures, sequencing and timing related constraints and information and design assumptions.

The system software (i.e. non-application software) should include software for diagnosing faults in the system hardware, error detection for communication links, and on-line testing of standard application software modules.

In the event of detecting an error or fault the system should, if appropriate, be allowed to continue but with the fault redundant element or complete part of the system isolated.

Detailed Design

The detailed design of the software modules and coding implementation should result in small manageable software modules. Semi-formal methods should be applied, together with design and coding standards including structured programming, suitable for the application.

The system should, as far as possible; use trusted and verified software modules, which have been used in similar applications.

The software should not use dynamic objects, which depend on the state of the system at the moment of allocation, where they do not allow for checking by offline tools.

ProSalus Limited

Slide 5 - 75



Functional Safety Engineering

Programming Language and Support Tools

The programming language should be capable of being fully, unambiguously defined and compliant with BS EN 61131-3. The language should be used with a specific coding standard and a restricted sub-set to minimise unsafe/unstructured use of the language.

The support tools need either to be well proven in use (and errors resolved) and/or certified as suitable for safety system application.

Software Module Testing and Integration

The individual software modules should be code reviewed and tested to ensure that they perform the intended function and by a selection of limited test data to confirm that the system does not perform unintended functions.

As the module testing is completed then module integration testing should be performed with pre-defined test cases and test data. This testing should include functional, 'black box' and performance testing.

The results of the testing should be documented in a chronological log and any necessary corrective action specified. Version numbers of modules and of test instructions should be clearly indicated. Discrepancies from the anticipated results should be clearly visible. Any modifications or changes to the software, which are implemented after any phase of the testing, should be analysed to determine the full extent of re-test that is required.

ProSalus Limited

Slide 5 - 76



Functional Safety Engineering

Overall Integration Testing

The overall testing of the integrated system includes both hardware and software, detailed requirement are in BS EN 61508-3 Annex A Table A.6 the same requirements are repeated in BS EN 61508-2.

The overall integration testing should be performed with pre-defined test cases and test data. This testing should include functional, 'black box' and performance testing.

The results of the testing should be documented in a chronological log and any necessary corrective action specified. Serial, Version numbers and test instructions used be clearly indicated. Discrepancies from the anticipated results should be clearly visible. Any modifications or changes to the system, which are implemented after any phase of the testing, should be analysed to determine the full extent of re-test that is required.

Validation

Whereas Verification implies confirming for each stage of the design that all the requirements have been met prior to the start of testing of the next stage, validation is the final confirmation that the total system meets all the required objectives and that all the design procedures have been followed .

ProSalus Limited

Slide 5 - 77



Functional Safety Engineering

IEC 61508 Part 3 Overview

Slide 5 - 78



Functional Safety Engineering

IEC 61508 Safety Certified PES Logic Solvers

- TUV Publish a list of type certified systems on website
- Ensure hardware and software versions are as per certificate
- Check Test report for any limitations on use
- Software representation complies with IEC 61131 requirements
- Within the use of LVL software there is the possibility to create user defined function blocks, however they must be constructed and tested as FVL software modules to avoid human or specification errors
- Certification can be directed at specific applications e.g. furnace control, HIPPS or for other typical process applications

ProSalus Limited

Slide 5 - 79



Functional Safety Engineering

IEC 61508 Software Verification

- Software verification complicated – 61508 requires formal analysis & traceability (61508-7 Annex D)
- Difficult and costly to test all foreseeable combinations of logic not normally considered in process applications reliance on SRS and C&E testing
- The failure modes are unpredictable in presence of hardware faults.
- Re-use of old software in new applications (also known as SOUP...software of uncertain pedigree - Refer HSE guidance RR 336/2001 & 337/2001

ProSalus Limited

Slide 5 - 80

Table A.1 – Software safety requirements specification (see 7.2)

TECHNIQUE/MEASURE *	REF	SIL1	SIL2	SIL3	SIL4
Computer-aided specification tools	B.2.4	R	R	HR	HR
Semi-formal methods	Table B.7	R	R	HR	HR
Formal methods	B.2.2, C.2.4	-	R	R	HR
<p>Note 1 – The software safety requirements specification will always require a description of the problem in natural language and any necessary mathematical notation that reflects the application.</p> <p>Note 2 – The table reflects additional requirements for specifying the software safety requirements clearly and precisely.</p>					
<p>* Appropriate techniques/measures shall be selected according to the safety integrity level. Alternate or equivalent techniques/measures are indicated by a letter following the number. Only one of the alternate or equivalent techniques/measure has to be satisfied.</p>					

Table A.3 – Software design and development: (see 7.4.4)

TECHNIQUE/MEASURE *	REF	SIL1	SIL2	SIL3	SIL4
1 Suitable programming language	C.4.5	HR	HR	HR	HR
2 Strongly typed programming language	C.4.1	HR	HR	HR	HR
3 Language subset	C.4.2	-	-	HR	HR
4a Certified tools and Certificated translators	C.4.3	R	HR	HR	HR
4b Tools and translators: increased confidence from use	C.4.4	HR	HR	HR	HR
<p>* Appropriate techniques/measures shall be selected according to the safety integrity level. Alternate or equivalent techniques/measures are indicated by a letter following the number. Only one of the alternate or equivalent techniques/measure has to be satisfied.</p>					

Table A.4 – Software design and development: detailed design (see 7.4.5 and 7.4.6)

TECHNIQUE/MEASURE *	REF	SIL1	SIL2	SIL3	SIL4
1a Structured methods	C.2.1	HR	HR	HR	HR
1b Semi-formal methods	Table B.7	R	HR	HR	HR
1c Formal design and refinement methods	B2.2.2, C.2.4	-	R	R	HR
2 Computer-aided design tools	B.3.5	R	R	HR	HR
3 Defensive programming	C.2.5	-	R	HR	HR
4 Modular approach	Table B.9	HR	HR	HR	HR
5 Design and coding standards	C.2.6, Table B.1	R	HR	HR	HR
6 Structured programming	C.2.7	HR	HR	HR	HR
7 Use of trusted/verified software elements (if available)	C.2.10	R	HR	HR	HR

* Appropriate techniques/measures shall be selected according to the safety integrity level. Alternate or equivalent techniques/measures are indicated by a letter following the number. Only one of the alternate or equivalent techniques/measure has to be satisfied.

Scope of compliance required for logic solver software products

- SIS Logic Solver and I/O certified for use at the relevant SIL
- All of the programming languages supported by the logic solver with any special safety functions and function blocks to be certified for compliance at the relevant SIL.
- All restrictions and operating procedures required by the certifying organization to be stated in the user documentation.
- Methodology for on-line testing using overrides to be approved by the certifying organization.



Functional Safety Engineering

Software Proven in Use - IEC61508-7 B.5.4 Field experience

For field experience to apply (*very difficult in reality – different firmware versions and missing FSM of the software*)

- unchanged specification;
- 10 systems in different applications;
- 100000 operating hours and at least one year of service history.

This documentation must contain at least

- the exact designation of the system and its components, including version control for hardware;
- the users and time of application;
- the operating hours;
- the procedures for the selection of the systems and applications procured to the proof;
- the procedures for fault detection and fault registration as well as fault removal.

ProSalus Limited

Slide 5 - 85




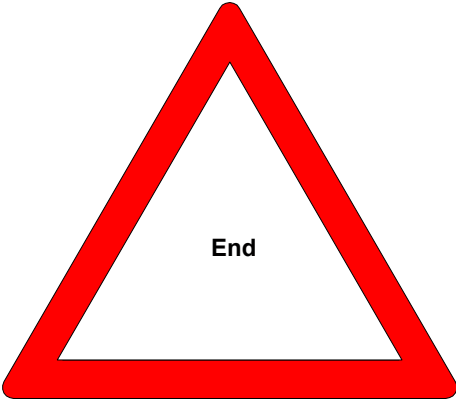
Functional Safety Engineering Summary

- Software safety integrity is achieved through IEC 61511-12 software life cycle and company software quality assurance procedures
- IEC 61508-3 is targeted at new PES devices but can be applied as necessary for end user support, but requires detailed knowledge
- Certified software packages provide a secure platform for the end user to execute an application.
- Vendor's training and safety manual requirements must be applied
- IEC 61511-2 Clause 12 provides additional support but is informative only

ProSalus Limited

Slide 5 - 86

 **Functional Safety Engineering**



End

ProSalus Limited Slide 5 - 87